PETITION TO MAKE SPECIAL	Docket No. C0012/7000

Applicant:	Raju C. Bopardikar, Jacob Y. Bast, Gary A. Cardone, David E. Kaufman, Stuart P. MacEachern, Bruce D. McLeod, James M. Nolan, Jr., Zdenek Radouch, Jack J. Stiffler and James A. Wentworth, III
Serial No:	09/608,521
Filed:	June 30, 2000
For:	METHOD AND APPARATUS FOR IMPLEMENTING HIGH-PERFORMANCE, SCALEABLE DATA PROCESSING AND STORAGE SYSTEMS
Examiner:	Not Yet Assigned
Art Unit:	2756

Box Petition
Assistant Commissioner for Patents
Washington, D.C. 20231

Applicant hereby petitions to grant the above-identified application special status in accordance with MPEP §708.02 VIII. In accordance with this request, applicant has performed the following:

(A) The petition fee set forth in 37 C.F.R. §1.17(I) is enclosed.

(B) Applicant believes that the present claims are directed towards a single invention, but will make an election without traverse, if necessary.

(C) A pre examination search has been conducted using the Delphion on-line database. The methodology of the search was to use selected keywords that are set forth below. Database hits were then examined, first by abstract, then completely, if deemed relevant. Patents discovered in the search which were deemed particularly relevant or relevant for back ground information were ordered. The keywords used and the results obtained as discussed in detail below.

(D) Copies of the references located in the search are attached.

(E) The first keywords used in the search were the names of several companies known to be working in related areas. These companies included Hitachi, Ltd., EMC Corporation, Auspex Systems, Inc. and Network Appliance, Inc. The following patents were discovered:

U.S. Patent No. 5,819,054 and continuation No. 6,012,119 (Ninomiya, et al.) disclose a storage system that is comprised of host devices connected to host adapter modules, storage devices connected to disk adapter modules and cache memories. The host adapters, disk adapters and cache memories are interconnected by a common bus to permit scalability. The host adapter modules, disk adapter modules and cache memories are duplicated and the common bus has two channels for fault tolerant operation. The host adapter modules, disk adapter modules and cache memories are also hot-pluggable so that the storage system can be maintained without shutting down the system. In operation, the host sends write information, including a physical write address and data, via the host adapter module, to the cache memory. The disk adapter modules read the information out of the cache memory based on the write information in the cache memory and store the information in the storage devices at the requested address (which may be reformatted to make it compatible with the storage devices.) For a read operation, if the requested information is in the cache memory, then it is read by the host adapter modules. Alternatively, if the requested information is not in the cache memory, then the disk adapter modules read the information from the storage devices and write the data and address information into the cache memory. Subsequently, the host adapter modules read the data and address information from the cache memory.

In a system constructed in accordance with the present invention, host devices (clients) are connected to access interfaces that, in turn, are connected to a pool of resources by a switch fabric. The access interfaces obtain information concerning resource availability from the resource pool and select the subset of the resource pool to use for any given transaction and to distribute the workload. Thus, the physical storage addresses at which data will be stored are selected by

the access interfaces rather than by the host devices. This selection makes it appear to each client that the resource pool is available to it without requiring the client to be aware that that the pool consists of multiple resources.

This structure is reflected in the claims. For example, claim 1 recites in lines 3-5:

> "an access interface module which receives requests for service from the client and selects a subset of the plurality of resources to provide the requested service and distribute the workload across the plurality of resources..." No such structure is taught or suggested by the Ninomiya reference.
>
> The remaining independent claims recite similar language as follows:
>
> Claim 12, lines 5-7
>
> Claim 36, lines 5-8
>
> Claim 44, lines 5-7
>
> Claim 55, lines 5-8
>
> Claim 79, lines 5-8

U.S. Patent No. 5,355,453 and continuations No. 5,802,366 and 5,931,918 (Rowe, et al.) disclose a file server architecture that has a network controller unit, a file controller unit, and a storage processor unit connected by a common bus. These units incorporate their own processors, and operate in parallel with a local host processor. The system is attached to a client via a network by the network controller unit, which performs most of the protocol processing. A virtual file system is implemented in the file control unit, and the storage processor provides high-speed multiplexed access to an array of mass storage devices. In operation, the client sends write information, including a file name, offset and count and data, via the network, to the network controller. The network controller extracts the write commands and data and forwards them to the file controller. The file controller converts the file name, offset and count information into disk sectors. The information is then passed to the storage processor which stores it in the specified disk sectors. For a read operation, the client specifies the file name offset and count, via the network. This information is processed by the

network and file controllers as with the write operation. The storage processor then retrieves the data at the specified disk sectors and passes it through the network controller to the client.

As previously mentioned in the present invention, access interfaces obtain information concerning resource availability from the resource pool and select the subset of the resource pool to use for any given transaction and to distribute the workload. Thus, the physical storage addresses at which data will be stored are selected by the access interfaces rather than by the client. The portions of the claims that recite this structure are identified above.

U.S. Patent No. 5,963,962 (Hitz, et al.) discloses a method for keeping a file system in a consistent state and for creating read-only copies of a file system. The patent discloses a write anywhere file layout (WAFL) which uses file metadata to convert between file names or specifications generated by a client to actual file locations in storage. When stored information is moved the metadata must be changed to indicate the new storage location of the file. The patent is concerned with maintaining the file system in a consistent state if a system failure should occur after file data has been moved but before the metadata has been updated with the new file information location. The disclosed method uses consistency points to insure that the system passes from one consistent state to another.

As with the previous two references, the Hitz reference does not disclose "an access interface module which receives requests for service from the client and selects a subset of the plurality of resources to provide the requested service and distribute the workload across the plurality of resources..." Locations in the independent claims where this language is located are described above.

In addition, several other keyword combinations were used during the search and resulted in patents considered relevant only as background information. A synopsis of these patents is given below.

Key word = "Storage system architecture" - 5 U.S. patents found, one relevant as background information:
5,634,033 -- Disk array storage system architecture for parity operations simultaneous with other operations

Key word = "Disk array architecture" - 2 U.S. patents found, both relevant as background information:
5,787,459 - Distributed disk array architecture - uses modular control units (MCUs) that communicate with both hosts and disks and, using a serial bus, with each other enabling a plurality of them to work as a unit.
5,479,653 -- Disk array apparatus and method which supports compound raid configurations and spareless hot sparing - describes a procedure for reconfiguration following a disk failure

Key word = "Data processing and storage systems" - 90 U.S. patents found; four relevant as background information:
5,239,640 -- Data storage system and method including data and checksum write staging storage
5,864,440 -- Data processing method and data storage system
5,809,543 -- Fault tolerant extended processing complex for redundant nonvolatile file caching - describes a disk caching scheme
5,787,304 -- Multipath I/O storage systems with multipath I/O request mechanisms - describes a method for connecting one host to multiple disk subsystems without having to synchronize writes for data integrity

Key word = "Fault-tolerant storage" - 14 U.S. patents found - two relevant as background information:

5,526,482 -- Storage device array architecture with copyback cache - uses a cache to support fast write acknowledgements; duplicates the cache if fault-tolerance is required

5,513,314 -- Fault tolerant NFS server system and mirroring protocol - pairs file servers to ensure file access in case of a failure of either server

Respectfully submitted

Paul E. Kudirka Date: 5/8/01

Paul E. Kudirka, Esq. Reg. No. 26,931
KUDIRKA & JOBSE, LLP
Customer Number 021127
Tel: (617) 367-4600 Fax: (617) 367-4656